Autômatos Finitos Determinísticos: Manual de Utilização

Júlio Zynger
110092466
julioz92@gmail.com

Sobre o projeto

Este projeto foi desenvolvido como trabalho final da disciplina “Linguagens Formais” do curso de Ciência da Computação da Universidade Federal do Rio de Janeiro, lecionada pelo professor Luís Menasché Schechter, no período entre 07/06 e 21/06 de 2011.

O projeto foi todo escrito sobre a linguagem Java (J2SE) e sua interface gráfica utilizou o framework Swing. O código-fonte é aberto e pode ser encontrado juntamente com o arquivo executável na minha página pessoal (http://dcc.ufrj.br/~julioz/projetos.html), ou via email (zynger@ufrj.br).

Funcionalidades

Seguindo a temática do projeto pedido pelo professor, o programa deveria ser capaz de:

1. Dada a descrição de um autômato finito determinístico e dada uma palavra no alfabeto do autômato, exibir a computação (sequência de configurações) completa realizada pelo autômato ao receber como entrada a palavra dada e informar se o autômato aceita ou não esta palavra.
2. Dada a descrição de um autômato finito determinístico A, determinar, através da aplicação do Algoritmo de Brzozowski, uma expressão regular r tal que L(r) = L(A).
3. Dada a descrição de um autômato finito determinístico A, determinar, através da aplicação do Algoritmo de Kleene, uma expressão regular r tal que L(r) = L(A).

A interface para entrada de dados

O programa foi desenvolvido com o framework Swing a fim de facilitar a entrada e edição de dados pelo usuário, e a visualização dos resultados.

Quando iniciado, o programa apresenta a seguinte tela:

Cada botão disposto nesta interface levará o usuário para a aplicação de uma das funcionalidades do programa (aceitação de palavra, aplicação de algoritmos: Brzozowski e Kleene).

Ao selecionar uma das opções, o usuário será levado para a tela de entrada de dados do autômato:

1. Campo para entrada dos símbolos do alfabeto do autômato. Os símbolos deverão ser separados por vírgulas (,).

2. Campo para entrada dos nomes dos estados. A adição do estado ao conjunto de estados só será feita após ser pressionada a tecla "enter", movendo então o estado para a caixa de texto logo abaixo.

3. Campo para entrada do nome do estado inicial do autômato. Neste campo deverá ser inserido o nome de um estado que já conste na lista de estados (item 2), caso contrário, o programa retornará um erro que impedirá a continuação da execução.

4. Campo para entrada dos nomes dos estados finais. Da mesma forma como no item 2, a adição de estados à este conjunto só será feita após ser pressionada a tecla "enter", movendo o estado para a caixa logo abaixo. No caso de adicionar estados à este conjunto que não estejam no conjunto de estados (item 2), o programa retornará um erro que impedirá a continuação da execução.

5. O botão "Transições" levará o usuário para a tela de inserção de transições entre estados (se nenhum erro descrito acima for detectado), que será da forma de uma tabela (estados x símbolos):

1. Lista de estados dos quais partirão as transições.

2. Lista de símbolos com os quais ocorrerão as transições

3. Ao clicar sobre uma célula em branco da tabela, um menu ‘drop-down’ será exibido, oferecendo ao usuário os estados para os quais a transição pode ocorrer. No exemplo da imagem, estaríamos escolhendo para qual estado a transição do estado q1 com o símbolo ‘0’ ocorreria.

4. Botão “Computar palavra” (ou “Executar algoritmo”, dependendo da funcionalidade tratada), executará a função do programa, e retornará o resultado da computação (ou a expressão regular do autômato).

5. No campo “palavra” (só está disponível quando o programa deverá computar uma palavra) deverá ser inserida a palavra a ser computada pelo autômato. No caso da palavra conter símbolos fora do alfabeto, o programa retornará um erro e não continuará sua execução.